



Τμήμα Μηχανικών Η/Υ & Πληροφορικής
Computer Engineering & Informatics Department (CEID)
www.ceid.upatras.gr

Εργαστηριακές Ασκήσεις

Εργαστήριο Σχεδιασμού VLSI

Υπεύθυνος:

Νικόλαος Σκλάβος,
Αναπληρωτής Καθηγητής

e-mail: [nsklavos AT upatras DOT gr](mailto:nsklavos@upatras.gr)

Πάτρα, 2020-21

Εργαστηριακή Άσκηση 01:

Εισαγωγή: Βασικά Στάδια

1. Ενότητες Εργαστηριακής Άσκησης

Ενότητα Α:

Εξοικειωθείτε με τη χρήση των εργαλείων:

- 1) Επεξεργαστές κώδικα γλώσσας περιγραφής υλικού VHDL *:

Ενδεικτικά παραδείγματα (όχι μοναδικά):

Linux, Mac:

Διαθέσιμη η έκδοση του Sublime Text Editor, δωρεάν στο σύνδεσμο:
<https://www.sublimetext.com>

Windows:

Διαθέσιμη η έκδοση του Notepad++, δωρεάν στο σύνδεσμο:
<https://notepad-plus-plus.org>

**: θα πρέπει να εγκαταστήσετε/ενεργοποιήσετε και την κατάλληλη μορφοποίηση (χρωματισμός σύνταξης κώδικα) για τη γλώσσα vhdl.*

- 2) Εξομοίωση, Σύνθεση:

Windows, κα:

Διαθέσιμη η έκδοση από το Xilinx Vivado, (University Program), δωρεάν στο διαδικτυακό τόπο:
<https://www.xilinx.com>

Linux, Mac, κα: (Μόνο εξομοίωση)

Διαθέσιμη η έκδοση του GTK Wave (Simulator), δωρεάν στο σύνδεσμο:
<http://gtkwave.sourceforge.net>

Αναφορά - Παράδοση:

-

Ενότητα Β:

1. Εξομοιώστε τον Κώδικα 1.
2. Επιβεβαιώστε τη λειτουργία που επιτελεί ο κώδικας.
3. Αναπτύξτε μια εναλλακτική μορφή, ισοδύναμου κώδικα. Πλεονεκτήματα και μειονεκτήματα εναλλακτικής μορφής.

** Κώδικας 1

LIBRARY IEEE;

USE IEEE.std_logic_1164.all;

ENTITY nand64 IS

PORT (inputA :IN std_logic_vector(63 downto 0);

inputB :IN std_logic_vector(63 downto 0);

outputC :OUT std_logic_vector(63 downto 0));

END nand64;

ARCHITECTURE structural OF nand64 IS

BEGIN

XORLoop: For i IN 0 TO 63 Generate

outputC(i) <= inputA(i) NAND inputB(i);

End Generate XORLoop;

END structural;

Αναφορά - Παράδοση:

- Τρόπος εργασίας για τα B1 και B2.

- Εναλλακτικός κώδικας και σύντομη εξήγηση του τρόπου σκέψης σας, για το B3.

Ενότητα Γ:

Να σχεδιαστεί η αρχιτεκτονική και να αναπτυχθεί σε κώδικα VHDL:

1. Ο πλήρης αθροιστής 1-bit, κρατούμενο εισόδου/εξόδου και άθροισμα για δύο ψηφία εισόδου.
2. Μια λογική μονάδων, δύο εισόδων (8-bit), η οποία υλοποιεί δύο λογικές πράξεις (AND, XOR) και παράγει δύο εξόδους (8-bit), μία για κάθε πράξη.
3. Στο κύκλωμα του Ερωτήματος Γ2, ενσωματώστε και ένα καταχωρητή για κάθε μια έξοδο.

Για όλα τα παραπάνω ερωτήματα, προτείνεται να δείτε πρώτα, τα παραδείγματα και τις ασκήσεις, που διδαχθήκατε στο θεωρητικό μέρος του μαθήματος.

Αναφορά - Παράδοση:

- Αρχιτεκτονική*, κώδικας VHDL και αποτελέσματα προσομοίωσης (εκτύπωση οθόνης, κυματομορφής/χρονοδιαγράμματος, βεβαίωσης ορθής λειτουργίας) για τα Γ1, Γ2 και Γ3.

*: Μπορείτε να χρησιμοποιήσετε προγράμματα για να σχεδιάσετε κυκλώματα/συστήματα, με έτοιμα σχηματικά όπως Visio, Concept Draw, OmniGraffle, ή κάποιο άλλο της επιλογής σας, PIN, αναπτύξετε τον κώδικα και ΟΧΙ ως αποτέλεσμα της σύνθεσης του κώδικα.

Εργαστηριακή Άσκηση 02:

Συνδυαστικά Κυκλώματα

1. Θεωρητικό Υπόβαθρο

Έννοιες – Λέξεις Κλειδιά Άσκησης:

- ✓ Συνδυαστικά κυκλώματα
- ✓ Τύποι δεδομένων εισόδου/εξόδου
- ✓ Διεργασία
- ✓ Αρχιτεκτονική δομής
- ✓ Κώδικας VHDL
- ✓ Προσομοίωση
- ✓ Αποτελέσματα προσομοίωσης

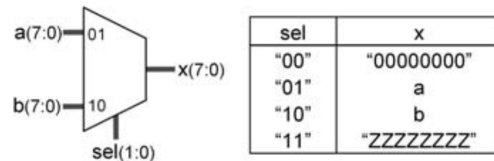
Εργαλεία Εργαστηριακής Άσκησης:

- **VHDL Editor** : Προτεινόμενοι, Notepad++, Sublime.
- **Xilinx Vivado** : Περιβάλλον εξομοίωσης.

2. Ενότητες Εργαστηριακής Άσκησης

Ενότητα Α:

1. Να αναπτύξετε τον αντίστοιχο κώδικα σε VHDL, για το παρακάτω κύκλωμα, σε δύο διαφορετικές εκδόσεις, με τη χρήση διαφορετικών βασικών εντολών κάθε φορά, (δύο διαφορετικά αρχεία κώδικα):



2. Μεταγλωττίστε και προσομοιώστε τον κώδικα σας. Επιβεβαιώστε, την ορθή λειτουργία του, με βάση τον αναμενόμενο τρόπο.

Παραδοτέο αναφοράς:

- Κώδικας VHDL, της Ενότητας.
- Αποτελέσματα προσομοίωσης (εκτύπωση χρονοδιαγράμματος οθόνης, κτλπ).

Ενότητα Β:

1. Υλοποιήστε με κώδικα VHDL την παρακάτω συνάρτηση:

$$Z = ABC' + AB + C$$

όπου Z: έξοδος, A,B,C: είσοδοι, εύρος ένα ψηφίο, για όλες τις εισόδους/εξόδους.

2. Για την παραπάνω συνάρτηση να δώστε δύο διαφορετικές προσεγγίσεις (δύο διαφορετικά αρχεία κώδικα): α) προσέγγιση με βάση διεργασία, β) προσέγγιση ως ένα πλήρες συνδυαστικό κύκλωμα (structural αρχιτεκτονική).
3. Μεταγλωττίστε και προσομοιώστε τον κώδικα σας. Επιβεβαιώστε, την ορθή λειτουργία του, με βάση τον αναμενόμενο τρόπο.
4. Εξηγήστε εάν υπάρχουν διαφορές ή όχι, συγκρίνοντας τις δύο προσεγγίσεις, καθώς και τους λόγους που οφείλεται αυτό.

Παραδοτέο αναφοράς:

- Κώδικας VHDL, της Ενότητας.
- Αποτελέσματα προσομοίωσης (εκτύπωση χρονοδιαγράμματος οθόνης, κτλπ).
- Απάντηση για το Βήμα 4.

Εργαστηριακή Άσκηση 03:

Σύνθεση - Μνήμες

1. Θεωρητικό Υπόβαθρο

Έννοιες – Λέξεις Κλειδιά Άσκησης:

- ✓ Σύνθεση
- ✓ Αποτελέσματα σύνθεσης
- ✓ Επιφάνεια – πόροι
- ✓ Απόδοση
- ✓ Μνήμες
- ✓ Ενσωματωμένες μνήμες σε FPGAs

Εργαλεία Εργαστηριακής Άσκησης:

- **VHDL Editor** : Προτεινόμενοι, Notepad++, Sublime.
- **Xilinx Vivado** : Εργαλείο εξομοίωσης και σύνθεσης.

2. Ενότητες Εργαστηριακής Άσκησης

Ενότητα Α:

1. Να υλοποιήσετε τη διαδικασία της σύνθεσης, για τους κώδικες των κυκλωμάτων-συστημάτων, που έχετε αναπτύξει στις Εργαστηριακές Ασκήσεις 1 και 2. Καταγράψτε με μορφή πινάκων τα αποτελέσματα για κάθε κώδικα, αναφορικά με τους δεσμευμένους πόρους – επιφάνεια και απόδοση: κρίσιμο μονοπάτι, συχνότητα λειτουργίας, συντελεστής απόδοσης κτλπ. Για τη σύνθεση να επιλέξετε το αντίστοιχο FPGA, που έχει τον πιο βέλτιστο συντελεστή κάλυψης, για το κάθε κύκλωμα-σύστημά σας, κάθε φορά.
2. Προσπαθήστε να εξηγήσετε αν τα αποτελέσματα του ερωτήματος 1, επιβεβαιώνουν τον αρχικό σας σχεδιασμό και απαντήστε εάν είναι αναμενόμενα ή όχι. Σε περίπτωση που δεν είναι αναμενόμενα, εξηγήστε το γιατί.

Παραδοτέο αναφοράς:

- Αποτελέσματα σύνθεσης (πίνακες μετρήσεων, σχηματικά σύνθεσης, κτλπ), ερώτημα A1.
- Θεωρητική απάντηση στο ερώτημα A2.

**** Παρατήρηση: Μην αλλάξετε του κώδικες, των Εργαστηριακών Ασκήσεων 1 και 2, ακόμα και εάν τα αποτελέσματα δεν είναι τα καλύτερα δυνατά.*

Ενότητα Β:

1. Υλοποιήστε με κώδικα VHDL μια μνήμη τύπου RAM, 32 θέσεων, με μήκος λέξης (θέσης) 8 ψηφίων. Μεταγλωττίστε και προσομοιώστε τον κώδικα σας. Επιβεβαιώστε, την ορθή λειτουργία του, με βάση τον αναμενόμενο τρόπο.
2. Αναζητήστε ένα διαθέσιμο τύπο FPGA, που να περιλαμβάνει ενσωματωμένη μνήμη. Ανατρέξτε στο αντίστοιχο εγχειρίδιο προδιαγραφών του, αναπτύξτε κώδικα VHDL, για την ίδια μνήμη με το ερώτημα 1, με τον τρόπο που περιγράφεται στο εγχειρίδιο. Μεταγλωττίστε και προσομοιώστε τον κώδικα σας. Επιβεβαιώστε, την ορθή λειτουργία του, με βάση τον αναμενόμενο τρόπο.
3. Να υλοποιήσετε τη διαδικασία της σύνθεσης, για τους κώδικες των κυκλωμάτων-συστημάτων, που έχετε αναπτύξει. Καταγράψτε με μορφή πινάκων τα αποτελέσματα για κάθε κώδικα, αναφορικά με τους δεσμευμένους πόρους – επιφάνεια και απόδοση: κρίσιμο μονοπάτι, συχνότητα λειτουργίας, συντελεστής απόδοσης κτλπ. Για τη διαδικασία της σύνθεσης, να επιλέξετε το ίδιο FPGA, (ερώτημα 2).
4. Εξηγείστε εάν τα αποτελέσματα του παραπάνω ερωτήματος, επιβεβαιώνουν τον αρχικό σας σχεδιασμό και αιτιολογείστε εάν είναι αναμενόμενα ή όχι.

Παραδοτέο αναφοράς:

- Κώδικες VHDL, της ενότητας.
- Αποτελέσματα προσομοίωσης (εκτύπωση χρονοδιαγράμματος οθόνης, κτλπ).
- Αποτελέσματα σύνθεσης (πίνακες μετρήσεων, σχηματικά σύνθεσης, κτλπ).
- Θεωρητική απάντηση στο ερώτημα B4.

Εργαστηριακή Άσκηση 04:

Συστατικά Στοιχεία – Σύστημα

1. Θεωρητικό Υπόβαθρο

Έννοιες – Λέξεις Κλειδιά Άσκησης:

- ✓ Πακέτο
- ✓ Συστατικό στοιχείο
- ✓ Γενικευμένη χαρτογράφηση
- ✓ Σύστημα

Εργαλεία Εργαστηριακής Άσκησης:

- **VHDL Editor** : Προτεινόμενοι, Notepad++, Sublime.
- **Xilinx Vivado** : Περιβάλλον εξομοίωσης και σύνθεσης

2. Ενότητες Εργαστηριακής Άσκησης

Ενότητα Α:

Επιλέξτε τέσσερις διαφορετικές πύλες, ψηφιακής λογικής σχεδίασης δύο εισόδων και μιας εξόδου με πλάτη:

- ✓ Πύλη Α, 8-bit.
- ✓ Πύλη Β, 16-bit.
- ✓ Πύλη Γ, 32-bit.
- ✓ Πύλη Δ, 64-bit.

Υλοποιήστε κάθε πύλη ως μιας ξεχωριστή ενότητα (entity σε ξεχωριστό αρχείο), με τη χρήση της γενικευμένης τεχνικής (generic), ως προς τα πλάτη, των εισόδων και των εξόδων.

Παραδοτέο αναφοράς:

- Κώδικες των τεσσάρων πυλών, σε VHDL, της ενότητας.
- Αποτελέσματα σύνθεσης για τις τέσσερις πύλες, (πίνακες μετρήσεων, σχηματικά σύνθεσης, κτλπ).

Ενότητα Β:

Δημιουργείστε μια μεγαλύτερη λογική μονάδα, που να περιλαμβάνει τις τέσσερις πύλες της παραπάνω ενότητας ως συστατικά στοιχεία (components), με τη χρήση της γενικευμένης χαρτογράφησης (generic map). Το σύστημα θα έχει δύο εισόδους και μία έξοδο των 120-bit, αντίστοιχα για το σύνολο των ψηφίων εισόδου και εξόδου, για τα εσωτερικά στοιχεία των τεσσάρων πυλών (8, 16, 32, 64 -bit).

Κατ' επιλογή του χρήστη θα εκτελείται από τη μεγαλύτερη λογική μονάδα, μόνο μια πύλη κάθε φορά και το αντίστοιχο αποτελέσματα θα προωθείται στα κατάλληλα ψηφία των εξόδων.

Παραδοτέο αναφοράς:

- Κώδικες VHDL, της ενότητας.
- Αποτελέσματα προσομοίωσης (εκτύπωση χρονοδιαγράμματος οθόνης, κτλπ).
- Αποτελέσματα σύνθεσης (πίνακες μετρήσεων, σχηματικά σύνθεσης, κτλπ).

Εργαστηριακή Άσκηση 05:

Σύνθεση υψηλού επιπέδου – Δημιουργία επιταχυντών

1. Θεωρητικό Υπόβαθρο

Έννοιες – Λέξεις Κλειδιά Άσκησης:

- ✓ Σύνθεση υψηλού επιπέδου - High level Synthesis (HLS)
- ✓ Ντιρεκτίβες – Directives/Pragmas
- ✓ Δομή ελέγχου ορθής λειτουργίας
- ✓ Βελτιστοποίηση σχεδιασμού

Εργαλεία Εργαστηριακής Άσκησης:

- **Xilinx Vitis HLS** : Περιβάλλον εξομοίωσης και σύνθεσης
- **Xilinx Vivado HLS** : Περιβάλλον εξομοίωσης και σύνθεσης

2. Ενότητες Εργαστηριακής Άσκησης

Ενότητα A:

Σε αυτή την εργαστηριακή άσκηση θα υλοποιήσουμε πολλαπλασιαστές πινάκων πρώτα σε VHDL και μετέπειτα σε γλώσσα C/C++ μέσω του Vivado ή Vitis HLS. Ο σκοπός είναι η εξερεύνηση του εργαλείου Vivado HLS/Vitis HLS και η κατανόηση των διαφορών στη διαδικασία σχεδίασης με το VHDL.

1. Να υλοποιήσετε έναν element-wise πολλαπλασιαστή δισδιάστατων πινάκων σε VHDL. Πιο συγκεκριμένα, υλοποιείτε την εξής πράξη:

$$\begin{bmatrix} a1 & a2 & a3 \\ a4 & a5 & a6 \\ a7 & a8 & a9 \end{bmatrix} \odot \begin{bmatrix} b1 & b2 & b3 \\ b4 & b5 & b6 \\ b7 & b8 & b9 \end{bmatrix} = \begin{bmatrix} a1*b1 & a2*b2 & a3*b3 \\ a4*b4 & a5*b5 & a6*b6 \\ a7*b7 & a8*b8 & a9*b9 \end{bmatrix}$$

Οι πίνακες θέλουμε να έχουν διάσταση 3x3 και τα στοιχεία των πινάκων A, B μέγεθος 5-bit. Βεβαιωθείτε ότι ο κώδικάς σας εκτελείται σωστά και έπειτα υλοποιείτε τη διαδικασία της σύνθεσης. Καταγράψτε τα αποτελέσματα.

2. Στη συνέχεια, ανοίξτε το Vivado HLS ή το Vitis HLS και υλοποιείτε την ίδια πράξη με τα ίδια χαρακτηριστικά σε C/C++. Χωρίς να προσθέσετε pragmas, εκτελέστε την σύνθεση του κώδικα και καταγράψτε τα αποτελέσματα. Επίσης, πραγματοποιείτε το βήμα του Co-Simulation και ανοίξτε το αρχείο του vhdl κώδικα που δημιουργήθηκε (project_name/solution1/impl/vhdl). Τι διαφορές βλέπετε σε σχέση με τον δικό σας vhdl κώδικα (ερώτημα A1);
3. Τέλος, προσθέστε τα pragmas που σας βολεύουν ώστε να ελαττώσετε όσο μπορείτε το latency και να επιταχυνθεί η υλοποίηση. Βεβαιωθείτε ότι ο κώδικάς σας δουλεύει σωστά υλοποιώντας διαδοχικά τις διαδικασίες του Simulation, Synthesis και Co-Simulation. Καταγράψτε τα αποτελέσματα του Synthesis και συγκρίνετέ τα με τον κώδικα του ερωτήματος A2. Εξηγήστε πως καταλήξατε στα συγκεκριμένα αποτελέσματα και την επίδραση των pragmas που χρησιμοποιήσατε. Επισημαίνεται τις διαφορές που βλέπετε σε αυτήν την vhdl υλοποίηση σε σχέση με τα προηγούμενα δύο αρχεία vhdl.

Παραδοτέο αναφοράς:

- Κώδικες VHDL και C/C++.
- Αποτελέσματα σύνθεσης για την VHDL, την C/C++ και την C/C++ με τα pragmas εκδοχή του κώδικά σας (πίνακες μέτρησης, σχηματικό σύνθεσης, latency, ποσοστό δεσμευμένων πόρων, κλπ).
- Θεωρητική απάντηση του ερωτήματος A2 και του A3.

*** Παρατήρηση: Επιλέξτε κάποιο τύπο FPGA που να έχει περίπου 360 DSPs. Είστε ελεύθεροι να επιλέξετε και κάποιον άλλο τύπο FPGA αρκεί να εξηγήσετε τον λόγο.

Ενότητα B:

1. Υλοποιείτε στο Vivado/Vitis HLS έναν πολλαπλασιαστή δισδιάστατων πινάκων. Συγκεκριμένα, υλοποιείτε σε C/C++ τον πολλαπλασιασμό δύο πινάκων διάστασης 3x3 με στοιχεία μεγέθους 12-bit. Βεβαιωθείτε ότι ο κώδικάς σας δουλεύει σωστά.

2. Προσθέστε τα κατάλληλα pragmas για να βελτιώσετε το latency και εκτελέστε την διαδικασία της σύνθεσης. Προσπαθήστε να κατεβάσετε το latency όσο μπορείτε. Εξηγήστε την επίδραση του κάθε pragma που χρησιμοποιήσατε και περιγράψτε την ροή εκτέλεσης του βελτιστοποιημένου κώδικά σας.

Παραδοτέο αναφοράς:

- Κώδικες με σχόλια.
- Αποτελέσματα σύνθεσης για τους δύο κώδικες.
- Θεωρητική απάντηση στο ερώτημα B2.